

CICE Consortium

Overview and Kick-Off Workshop Report

December 2016

Executive Summary

CICE's community of stakeholders and primary developers has formed a new sea ice modeling consortium as a vehicle for collaboration and sharing, to maintain the current CICE model for existing and new users, to incorporate and maintain new research and development, and to accelerate scientific sea ice model development and its transfer into operational use.

Development and maintenance of the Los Alamos sea ice model, CICE, has been led and coordinated by the Department of Energy since the early 1990s. Over the past two decades, a broad community of climate and weather forecasting groups has adopted the code. As the Department of Energy continues to develop a next generation of climate models, continued support of CICE is needed while the community continues to improve and use the model for their own applications.

Although the initial focus of the Consortium is on the CICE sea ice model, there is a desire to continue our collaborations in the longer term, beyond the life of CICE. Therefore the Consortium will be set up as a framework able to evolve with general sea ice model contributions in the future.

This report documents the decisions and recommendations made during the Consortium's initial workshop, held October 26-27, 2016, in Santa Fe, New Mexico. Participants included representatives from the U.S. Department of Energy, the U.S. Department of Defense, the National Oceanic and Atmospheric Administration, the National Science Foundation, Environment and Climate Change Canada, and the UK Met Office. A complete list of participants and their affiliations can be found in the appendix.

General goals for the workshop encompassed elucidation, discussion, and clarification of participants' expectations for the Consortium, their requirements for both the Consortium and the code itself, a governance structure for the Consortium including decision-making and code management processes, the responsibilities of Consortium Members, model users and code contributors, and an initial implementation plan.

Although workshop participants recommended that the Consortium be designed for long-term flexibility as their needs and the code evolve, the funding sources for their initial commitments limit the timescale for setting up the Consortium and getting it running to 2 years.

Nomenclature: “Icemark” refers to the column physics (ice thickness distribution, ridging, thermodynamics and biogeochemistry), and “CICEdyn” to the dynamical core (momentum, rheology and transport modules). The full model is “CICE”, and “CICEcore” refers to everything in the model except Icemark.

Major recommendations

For long-term flexibility, the software for CICEdyn and for Icemark should evolve into flexible, interoperable libraries. For example, Icemark includes the vertical physics, i.e. limited to a single grid cell, and can be utilized in other sea ice models that are based on unstructured grids. Therefore the Icemark modules should be separated from the rest of the CICE model for ease of use and testing.

A thorough test suite should be developed for both CICEdyn and Icemark, along with CICE as a whole, and be distributed with the model software. This testing mechanism would be separate from other testing done by individual users or centers in their own model configurations that use CICE.

Post-processing tools should be made available, including a benchmarking suite for evaluating the model output both in stand-alone mode and in coupled configurations.

Updates to the code should be released (to beta testers at least, or to everyone) on a regular, frequent schedule to encourage all developers to work with an up-to-date version. The recommendation was for twice-yearly releases to beta testers and formal public releases roughly every 2 years, to coincide with major additions or advances in CICE code. A formal agreement should be developed for a small group of alpha testers from institutions that are not Consortium Members, to access to the most up-to-date version possible.

General support for the current release should be provided, and only bug fixes for the previously released version.

GitHub should be the platform used for the repository. Consortium Members require flexibility for code sharing, with an open-source repository for the primary branches (master/beta versions) but allowing private branches for development.

A paid software engineer or “code manager” should be an integral part of the Consortium’s operations and oversight.

The governance structure and requirements should not be overly burdensome for Consortium Members (e.g. reporting) or for community involvement (e.g. ease of code access to encourage new development and involvement of early career scientists). Participants recommended limiting formal reporting requirements to no more than once per year.

The overall governance structure should consist of an Executive Oversight Board, a Lead Coordinator and Co-lead Coordinator, a Software Engineer, and Task Teams with Team Leads. Some Task Teams should have Co-leads for contingencies, while other Teams have strong support Consortium-wide. There will also be non-Consortium developers and users of the code. A committee of Team Leads will act as a science oversight and advisory board for future planning, and function as a change control board for new code developments. The Executive Oversight Board will communicate and coordinate with a committee of senior stakeholders (“Sponsors”) to ensure support for the Consortium is maintained at an appropriate level. Workshop participants nominated Executive Oversight Board representatives and volunteered for Team Lead positions, as detailed below.

A Co-lead Coordinator should be identified to share the workload of the Lead Coordinator and to “learn the ropes” for possible succession to the Lead Coordinator position. Elizabeth Hunke is recommended as the initial Lead Coordinator, a role she expects to fulfill for no more than 2 years. It would be preferable for her successor to have the facilities to run the entire CICE model in stand-alone mode for testing and coordination with all other developers.

Decision-making authority begins at the Team level, with increasing coordination via Team Leads and the Lead Coordinator as the issue at hand involves more Teams.

For Consortium Membership, initial contributions are at least 0.25 FTE, but this requirement may be relaxed later as the workload decreases after initial Consortium setup, and in order to entrain newer members of the R&D community.

Consortium Members (Team Leads) should prioritize future plans, considering the scientific and operational needs of Consortium Members and the larger community. For instance, a Data Assimilation task (or Team) should be introduced for broader sharing of this code capability. Planning horizons should be commensurate with proposal/project cycles (approximately 3 years).

Finally, after taking a vote, the proper Pronunciation of “CICE” was deemed to be ‘sice’ (as pronounced by the British), although the letters may be spelled out for complete clarity. Pronouncing the name as “sea ice” (as originally intended) now is judged to be spoken with an American accent.

Table of Contents

Executive Summary	1
Major recommendations	2
Introduction and Vision	5
Brief History of CICE Milestones and Collaborations	5
Expectations	6
The future of community sea ice model collaboration	6
Research and operational drivers	7
Participating Organizations	7
Requirements for the Consortium	9
Requirements for the code	10
<i>General code requirements</i>	
<i>Stand-alone configuration</i>	
<i>Coupled configurations</i>	
<i>Icepack</i>	
<i>Testing and Analysis</i>	
Requirements & Recommendations for Code Licensing and the Repository	11
<i>Licensing</i>	
<i>Repository</i>	
<i>Code redistribution</i>	
Implementation	13
Consortium Governance	13
<i>Roles</i>	
<i>Consortium Membership</i>	
<i>Policies and guidance</i>	
Code Management	20
<i>Constraints</i>	
<i>Process</i>	
<i>Code releases</i>	
<i>Confidence Testing</i>	
<i>The principle of “Do no harm”</i>	
Next Steps	24
Consortium organization	24
Code	25
Initial Timeline	27
Appendix A: Acronyms	28
Appendix B: Participant List	29
Appendix C: Governance/Process FAQ	30

Introduction and Vision

Brief History of CICE Milestones and Collaborations

Development of the CICE model began at Los Alamos National Laboratory (LANL) in 1994 with a translation of the Semtner (1976) 3-layer thermodynamic code into Fortran 77. Development of the Elastic-Viscous-Plastic (EVP) sea ice dynamics model followed, which attracted the first community user for the code, the Naval Postgraduate School (NPS). CICE v1 was released in 1998 and initial attempts to validate the model ensued, in collaboration with researchers at the Naval Research Laboratory – Stennis Space Center (NRL-Stennis). The Naval Postgraduate School again supported the EVP model's advance by implementing it in the DOE-supported Parallel Climate Model (PCM), based at the National Center for Atmospheric Research (NCAR). Development efforts for PCM and NCAR's other climate model, CSM, later merged to become CCSM and adopted CICE v3 in 2002. A formal Memorandum of Understanding between LANL and NCAR formed the foundation for a strong and lasting collaborative relationship for sea ice model development, verification and validation, and scientific research. NOAA's Geophysical Fluid Dynamics Laboratory (GFDL) later adopted EVP for use in their CM2 climate model, leading to its use in the National Centers for Environmental Prediction (NCEP) numerical weather prediction model CFS in 2011. The UK Met Office (UKMO) adopted CICE v3.1 in 2005 for use in its climate model and later for the full range of its "seamless prediction system" climate and weather prediction models. The coupling scheme used by UKMO is different than that used in other climate and operational models, requiring oversight for subsequent community contributions to the code to ensure that both coupling mechanisms continue to work properly. Canadian research groups began adopting the CICE model during the late 2000s, and in 2009 NRL-Stennis implemented CICE v4 in their Arctic Cap Nowcast-Forecast System (ACNFS) for operational use by the U. S. Navy and other federal centers. Following several years of validation and acceptance testing, ACNFS became operational in 2013, incorporating CICE v4.0.

Meanwhile, the Model for Prediction Across Scales (MPAS) framework for climate model components was developed, spurring discussions about an MPAS-seaice model in 2011. MPAS provides a framework of operators and code infrastructure for component model development on variable resolution, unstructured grids. A design document for MPAS-seaice was prepared in 2012, with funding for the project included in DOE's Accelerated Climate Model for Energy (ACME) project beginning in 2014. Because MPAS-seaice utilizes an unstructured grid, the grid-dependent portions of CICE had to be recoded. Physical parameterizations that are limited to single grid-cell columns could be shared between the models, however, and these modules (the "column package", now renamed "Icepack") were rewritten to make them independent of the grid and other CICE infrastructure. CICE v5 was released in 2015 without the column package changes; a formal agreement between the ACME project and CICE's community users enabled them to access and use this version of the code prior to its release.

With the exception of the LANL-NCAR Memorandum of Understanding, collaborations among other members of the CICE community have been informal. Updated versions of CICE have been released every 2 to 3 years, enabling other research groups to utilize and develop the model. The collaborating institutions listed above and many of the other community research groups have contributed code (bug fixes, new parameterizations, infrastructure, etc.) back to the central CICE repository at LANL, coordinated and overseen by Elizabeth Hunke with support from DOE.

A more formal arrangement for sea ice modeling collaboration is needed for several reasons. First, significant multi-decadal changes in both the Arctic and the Antarctic require improved understanding of the role of sea ice in the earth system. CICE is also used to meet civil and military operational prediction demands and to provide seasonal to centennial climate predictions. Sea ice modeling efforts at DOE and in the larger community have benefitted greatly from many CICE modeling collaborations, a community success story that should be sustained. Finally, DOE's shift to the MPAS framework for ocean and sea ice models, which led to creation of the column package (Icepak), opens up additional opportunities for other groups to utilize this portion of CICE in their own sea ice models. DOE is interested in sharing responsibility, coordination and support of community sea ice modeling efforts with the community, and is therefore responding to the community's suggestion for the formation of a sea ice modeling consortium.

Expectations

The CICE Consortium is intended

- To be a vehicle for collaboration and sharing
- To accelerate scientific development and its transfer to operational use
- To maintain the current CICE model for existing and new users
- To incorporate and maintain new research and development but not perform the R&D itself
- To attract multi-center buy-in with a common vision and direction
- To incorporate a governance structure and decision-making process
- To be fully funded through in-kind contributions
- To provide value to Consortium Members at least equal their investment
- To provide sea ice model testing and analysis tools to the public
- To support the CICE sea ice model and its descendants, but not the coupled models or configurations that use them

The future of community sea ice model collaboration

Workshop participants expressed a strong desire to continue sea ice modeling collaborations into the future, beyond the potential lifetime of the CICE model. Many of them expect to move to other modeling frameworks, perhaps using only Icepak or other capabilities such as the analysis tools, and they would like to

continue to share their new developments across the community, even when the developments are not specific to the CICE model. This Consortium will be designed to fill that need in the longer term, but initially it will focus on the CICE model in order to build the foundation for future expansion.

Research and operational drivers

Over the past two decades, a broad community of climate and forecasting centers has adopted the CICE code. The model is used for research and development purposes as well as operational forecasting, which put different constraints on their modeling needs. As a result, there are particular requirements for how the Consortium is organized and run, as well as for the code itself and the shared repository in which the software and documentation will be kept.

Participating Organizations

One representative from each of the agencies or institutions at the workshop gave an introductory presentation explaining their primary uses of the CICE model, key requirements, and potential contributions to the Consortium's work.

DOE – As mentioned above, DOE's historical role has been model development and support. DOE is now interested in continuing its community coordination role for the model, particularly Icepack. Many of DOE's requirements are already incorporated in the code, because of its historical role as the primary model developer. New requirements for Icepack include (1) a clean interface for linking with other models, (2) software best-practices including design documents, model requirements, developer's guide, explicit coding standard, unit tests, column model tests, etc., (3) a repository structure such that Icepack can be incorporated and developed easily/efficiently in other sea ice models (e.g. git submodule separate from the rest of the CICE model), and (4) the option to maintain new developments on a non-public repo branch until they are ready for release. DOE offered to serve as Team Lead for Icepack, make contributions to the biogeochemistry and thermodynamics modules within Icepack, and fill the Consortium Lead Coordinator role for 1-2 years (0.5 FTE).

NCAR – NCAR has long led a community climate modeling effort with CESM. They support a number of climate science goals through a single model code base, and while the set of supported configurations is limited, it is still quite large with 3 resolutions; coupled, stand alone, and greenhouse gas configurations, etc. They have contributed a number of science developments and code infrastructure improvements to CICE, and they provide community support such as diagnostic and post-processing tools, community workshops (Polar Climate Working Group meetings occur twice a year), bulletin board and tutorials, and a community liaison. They require a state-of-the-science model that is well documented, conserves heat and water, is suitable for varied climate regimes (not tied to present day observations), is computationally efficient, and enables community involvement.

They offered to fill the Consortium Team Lead role for community support and also make contributions to testing, infrastructure, and analysis (1.0 FTE).

NPS – NPS leads the Regional Arctic System Model (RASM) effort, which includes 8 collaborating institutions. Their science focus is on high-resolution Arctic simulation; NPS is specifically interested in sea ice dynamics. Over the next few years they will test the current model and new contributed physics in their coupled configuration. They also provide community support for code releases and bug tracking. Roles for the Consortium can include Team Lead for CICEdyn, especially the transport tasks, thermodynamics and ridging/morphology tasks for Icepack, and some support for the output infrastructure (0.25 FTE). In the future NPS expects to contribute a satellite emulator to the analysis tool set.

NRL-Stennis – The U.S. Navy uses CICE as part of their prediction capability to forecast ice cover over the Arctic and Antarctic, which is a coupled ocean-ice modeling system that assimilates both ocean and ice data. The model resolution is currently 3-4 km, moving to 2 km. The Navy currently forecasts sea ice conditions over 7 days, and they are moving to longer timescales. As part of the Navy's ESPC forecast products, CICE output is sent to the National Ice Center (NIC) and to NOAA daily. The Navy works with NIC to develop very specific products for them. NRL-Stennis also contributes to the Sea Ice Prediction Network. For the Consortium, they expect to serve as Team Lead for the Testing and Analysis Team, contribute code and support for data assimilation, and also help with code infrastructure tasks (0.5 FTE).

UKMO – The UK Met Office is developing a “seamless forecasting system”, the Unified Model, in which the same code is used for a variety of spatial and temporal scales ranging from short-term forecasting to climate projections. Their global partnership for weather and climate applications has partners in Australia, India, New Zealand, South Korea and the UK. To encourage community collaborations, they have instituted the Met Office Science Repository Service, in which they plan to deposit CICE. The UK's CPOM group has contributed high-resolution, high fidelity models of melt ponds, an anisotropic sea ice rheology, and a form drag scheme to CICE. Their coupling methodology is different from that used in CESM and other models. UKMO originally offered the repository Team Lead role to the Consortium, pending further discussion of repository design and agreements. They will co-Lead the Community Support Team and contribute to the data assimilation effort (0.5 FTE plus possibly an additional 0.5 FTE from CPOM at the University of Reading).

ECCC – Canadian modeling groups are responsible for short-term and seasonal forecasting around Canada, and for climate simulation. Their current efforts include developing a grounding scheme for modeling landfast ice, and an implicit solver for the momentum balance. For the Consortium, ECCC can serve as the Team Lead for CICEdyn with momentum/rheology tasks (0.25 FTE). They also offered to design a logo for the Consortium.

NOAA/GFDL – Scientists at the Geophysical Fluid Dynamics Laboratory have not been CICE developers or users (except EVP) in the past, and now they would like to incorporate Icepack into their sea ice model (SIS). They require open source development; a collaboration among GFDL, LANL and NCAR scientists to develop the CVMIX parameterization for the ocean model may serve as a model for a similar, broad Icepack collaboration in the Consortium. They offered to co-lead the Icepack Team (0.25 FTE).

NOAA/NWS (and others) – The National Weather Service’s current prediction suite needs to be consolidated into a set of products serving well defined requirements. In particular, they need an advanced physics sea ice model. They have been working with the Navy to obtain sea ice simulation products from CICE. On daily time scales they need to cover the Great Lakes and the Arctic. NOAA would like the column physics and grid-dependent code to be separable, preferably as a framework with exchangeable grids. They desire a true community modeling framework including contributions and maintenance, funded through in-kind contributions. NWS/OSTI can support a dedicated code manager/software engineer who would also serve as Team Lead for the Infrastructure team (0.5 FTE).

Requirements for the Consortium

- Communication and coordination linkages are critical and must be robust.
- Implementation of the Consortium’s plans must be phased in, prioritizing existing code functionality and required testing above new additions to the code.
- Prioritization of future plans shall consider the scientific needs of all Consortium Members and the community.
- The Consortium design must make sense for both research and operational work.
- Namelist settings and other documentation for validated configurations should be deposited in the repository, so that other users and developers do not need to comb through the literature to find the details needed to reproduce validated configurations.
- Regression testing must be simple and easy for everyone to perform. Modeling centers will be requested to also regression-test the code in their own configurations.
- A policy or process must be developed for invasive changes to the code, when new, answer-changing code contributions cannot be turned off via namelist or other flags.

- If git is used for the repository, svn access must be optionally available.
- A code manager or software engineer is needed for gatekeeping of code changes. By gatekeeping, we mean making sure that code changes meet a set of prerequisites for committing code to the CICE repository, as discussed in this plan.

Requirements for the code

In the past, the ability to share new code developments through the CICE repository has been a useful selling point for funding proposals. To encourage such community contributions, adding new code should be allowed as long as it satisfies the agreed-upon software best practices (to be determined), but do not allow multiple parameterizations that are very similar, i.e. avoid duplicative effort and code.

General code requirements

- Conservation of heat and moisture to machine precision
- Flexibility for use in different coupled model configurations
- Suitability for other climates (e.g. paleoclimate), ice regimes (e.g. lake ice, other planets), various time scales (especially short term predictions), and various resolutions (especially high-res)
- Software best practices
- Modularity with clean interfaces
- New contributions have an on/off option if possible (“do no harm”)
- Repository branches must be synchronized frequently with the trunk/master
- Well documented code and instructions for performing basic testing and analysis
- Optimized software (without degradation of physical accuracy)
- Release versions and regularly tagged versions of the trunk remain frozen (uneditable) so that the exact code can be reproduced and used as needed for both science and code development.

Stand-alone configuration

Many institutions (including Consortium Members) are not using CICE as a stand-alone model, but it is in the Consortium’s interest to collectively maintain a version of the model that is stand-alone as a common testbed and a means to bring others into the effort.

Coupled configurations

The responsibility for maintaining coupling interfaces between CICE and their coupled models lies with individual modeling centers, but the ice-side interfaces for these configurations can be kept within CICEcore or Icepack repositories. Testing and validation of these interfaces should be completed by individual centers in their coupled configurations, and, where possible, the results of these tests made available to other CICE Consortium members to help build confidence in code versions.

Icepack

The Icepack code and repository structure needs to be flexible for groups to easily use the package with various dynamical cores. In particular, Icepack needs to reside in a repository that is distinct from other CICE code. We anticipate the rest of the CICE code will reside in the CICEcore repository. The Consortium must maintain a set of ice-side interface modules for Icepack suitable for use in various other model configurations (e.g. DOE, GFDL), and a driver with specific tests for Icepack needs to be developed.

Testing and Analysis

Testing and analysis tools have not been released with CICE in the past, due to the extra effort needed to maintain them for the community. The Consortium cannot test all combinations of physics settings or other modifications, including different namelist combinations, but stand-alone CICE and Icepack tests should be provided for a pre-defined set of options.

A testing mechanism should be employed that builds and tracks confidence in each new model version, as the model passes basic and more realistic tests in multiple coupled model configurations.

An analysis package needs to provide new and varied metrics for community use, including a benchmarking suite for initial evaluation of code changes. (Benchmarking in stand-alone mode does not guarantee that the code will perform well in a coupled model.)

Workshop participants agreed that further discussion is needed to determine requirements governing the scope of testing provided by the Consortium, for data assimilation and metrics, and what code configurations will be supported (e.g. C grids). Developing appropriate tests will be an iterative process.

Requirements and Recommendations for Code Licensing and the Repository

Consortium Members have various requirements for code sharing, licensing and intellectual property. Policies addressing these needs must be developed, with some initial requirements or restrictions noted below.

Licensing

A license for use of any part of the CICE software package should be developed. There are several examples of intellectual property policies and agreements that can be considered, e.g. CESM, ACME, CVMIX, and NOAA's WaveWatch3 license.

Consortium Members presented a variety of perspectives on these issues:

1. DOE/LANL currently has a copyright statement that must be distributed with the code.

2. GFDL's models are open source.
3. NOAA/NWS is allowed to distribute their models piece by piece, but the unified weather prediction model does not become open source.
4. NCAR/CESM has an agreement that deals with intellectual property rights.
5. The Navy must deal with national security issues, but they have not had any issues using CICE so far.
6. Anyone can use the UKESM for research purposes within the UK.

Repository

Consortium Members will need to abide by a joint agreement regarding open source and private development branches. For instance, the model itself may be open source, but particular configurations are used by modeling centers. In particular, Icepack must be completely open source for GFDL.

A central location for the sea ice model and associated software and documentation, such as shared test cases and analysis tools, is preferred. Many institutions already have their own CICE repositories, which are linked to the primary CICE repository at LANL through both manual and automated methods.

Hosting the repository on Github has a number of virtues and is recommended, pending further investigation by a (temporary) Repository Committee formed during the workshop.

1. Git can be used to access the repository, with optional access via subversion (svn) commands.
2. Tools for automated testing and bug tracking software are available.
3. Repository access can be made seamless for separate pieces of code (e.g. Icepack, CICEdyn, Infrastructure, the Testing and Analysis suites, and validation documentation could each be held in distinct repositories under a Consortium umbrella).
4. Initial development can be performed on private branches while allowing open-source access to other users.

Code redistribution

Consortium Members currently handle redistribution of the code differently. For instance, CICE is not redistributed by UKMO – their users must get the code from the primary CICE repository and then apply UKMO's changes as needed. NCAR redistributes released versions of CICE with CESM. A uniform policy should be developed.

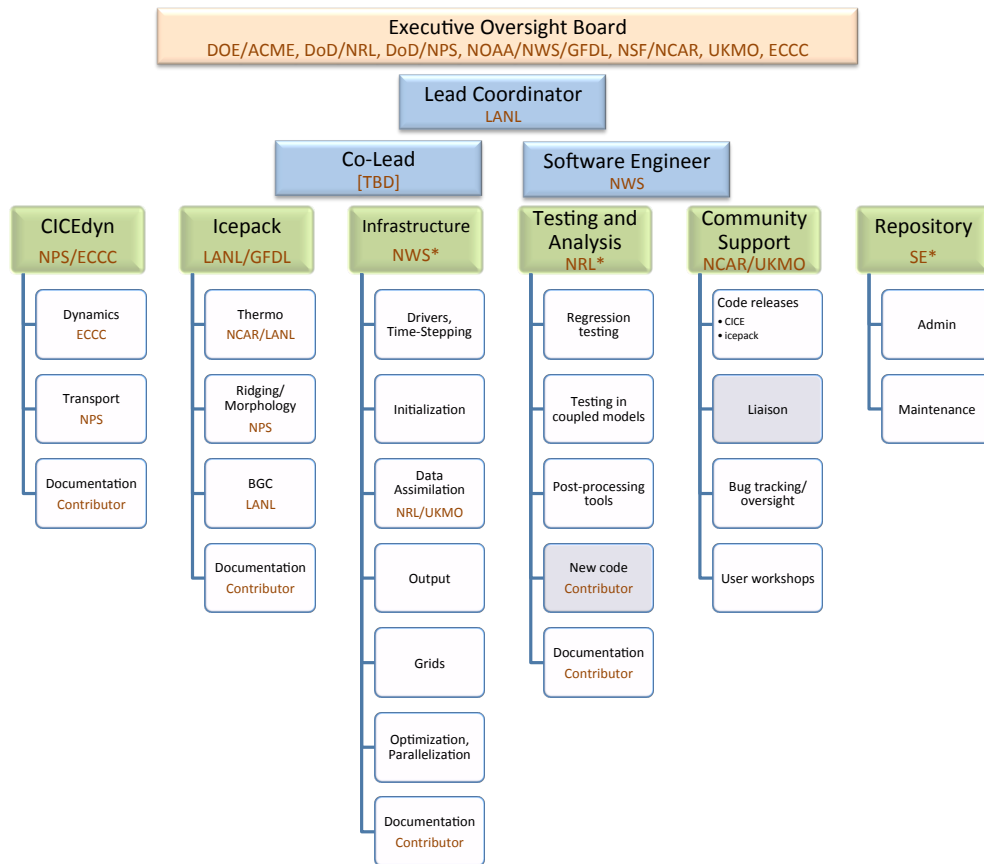


Figure 1. Org Chart. White boxes indicate task areas or responsibilities for each of the six Teams, including code elements, post-processing software, and community interfaces. Member institutions responsible for tasks or teams are shown in red. * indicates that (nearly) all of the Consortium Members are interested in that team's activities, may already be doing these tasks for CICE at their institutions, and will assist the listed lead institution and/or participate directly in those activities.

Implementation

Consortium Governance

The governance structure proposed and discussed during the workshop, detailed below, is designed to boost efficiency while still providing oversight and a decision-making process. Generally speaking, decision authority resides with team members doing the work, with more communication, coordination and governance structure required for decision-making as cross-team impact increases. Governance requirements can be relatively informal for Teams (e.g. guidelines); greater governance is needed for agreements with the broader community. Higher levels of Consortium management will handle the reporting requirements, with input from the Team Leads. Only Consortium Members' representatives have decision-making authority at any level within the Consortium.

Figure 1 contains the Organizational (Org) Chart proposed for the Consortium structure, showing the Executive Oversight Board in tan, Lead and Co-lead Coordinators and the Software Engineer in blue, and the six Teams with Team Leads in green.

The higher-level, colored boxes indicate more of an oversight or coordinating role. Responsibility for the “hands-on” work resides in the white task boxes and includes consulting with contributors, testing changes, reading code, looking for efficiency gains, improving user friendliness, etc. for code changes generated within the Consortium structure (e.g. bug fixes) and for contributed code. The Lead Coordinators and Team Leads should also be involved with the code, both in an advisory role (to make sure the changes don’t break the model for other teams or the rest of the community) and directly (the more eyes, the better).

New code contributions must have already undergone testing, but coordination with other Teams should already also have occurred during the development process, often initially coordinated through the community liaison (grey boxes). The Community Support Team will need to work with all Teams.

The committee of Team Leads includes Lead and Co-lead Coordinators and the Software Engineer. Nominations for leadership positions are provided bottom-up.

Roles

Sponsors (not shown on the org chart) are responsible for agency-level oversight, policy authority on behalf of their organizations, and coordination of resources. Members are agency representatives, e.g., DOE, NOAA, US Navy, NSF, UK Met Office. Sponsors meetings will be convened as needed and chaired by Dorothy Koch, DOE.

Executive roles:

- Agree on the financial commitment of Consortium Members for the next financial year
- Appoint Executive Oversight Board members
- Decide whether to include or expel any new or existing Consortium Member
- Settle disputes among Consortium Members

Advisory roles:

- Advise on funding issues and opportunities relevant to continued progress
- May review Consortium plans and progress

The **Executive Oversight Board** (EOB) is responsible for strategic direction, general oversight of software/documentation support levels, and coordination with Sponsors to ensure financial support is maintained at an appropriate level for the Consortium. Board members are institutional or project level management

representatives, who are appointed by and report to Sponsors. Each Sponsor appoints at most two representatives.

Executive roles:

- Appoints the Lead Coordinator and Task Team Leads
- Communicates objectives and constraints to the Lead Coordinator
- Approves a Consortium [frequency] Plan developed by the Lead Coordinator and Team Leads
- Draws up (or delegates) Consortium agreements with sea ice model developers or users, specifying expectations and requirements for model use prior to public release
- Approves any modifications needed to Consortium agreements
- Becomes involved in science decisions only when there is an unresolved dispute
- May facilitate external reviews
- Approval of this Board is required if the answer to any of the following questions is yes and the committee of Team Leads has not reached consensus for a solution:
 - *Does the change have resource or funding implications for the Sponsors?*
 - *Will the change significantly affect backwards compatibility?*
 - *Will the change significantly affect Consortium Members, users or developers?*
 - *Does the change significantly shift the functionality and scope of CICE?*
 - *Are there any licensing issues?*
 - *Are there any security concerns that need to be addressed?*

Advisory role:

- Reports to Sponsors, including resource needs

“Consortium Members” are the institutions or projects represented on this Board, making substantial in-kind contributions toward achieving the Consortium’s goals and purposes. Consortium Members may have alpha testers who are in other institutions (e.g. CPOM for UKMO), approved by the EOB, who have access to the Consortium’s most recent master code branch.

The **Lead Coordinator** is responsible for overall coordination and management of the Task Teams, in consultation with the Executive Oversight Board, and is appointed by and reports to the Executive Oversight Board.

Coordination roles within the Consortium:

- Coordinates among Executive Oversight Board and Task Team Leads
- Chairs meetings of the Task Team Leads
- May overrule decisions made by Task Team Leads

- Clarifies scientific and technical priorities
- Monitors progress against plans
- Ensures timely and appropriate reviews of proposed code contributions
- Delegates tasks as needed
- Reports periodically to the Executive Oversight Board
- Attends Executive Oversight Board meetings

Planning and Outreach roles:

- Leads development of a Consortium [frequency] Plan that includes goals, tasks, milestones and required resources
- Organizes periodic user workshops to provide a relaxed, informal forum for presentation of recent progress, discussion of problems and constructive feedback from users, and to facilitate the development of collaborations between groups
- Maintains the visibility of the model and software before the climate science community and other such organizations as necessary

The **Co-Lead Coordinator** assists the Lead Coordinator as needed for all activities, is included in meetings of the committee of Team Leads, and is potentially a successor for the Lead Coordinator.

The **Software Engineer** provides broad oversight of the code to ensure compatibility and software best practices, and software consulting support for team members as needed. The Software Engineer will coordinate closely with the Lead Coordinator and Team Leads, and generally speaking will not decide what scientific capabilities go into the code, but rather ensure that the software works. The Software Engineer will participate in meetings of the committee of Team Leads. An early-career technician or scientist could be added to help with code management tasks. Particular SE roles include

- coordinating changes to code infrastructure,
- assisting in the design of the testing framework, including subcomponent functionality testing
- verifying contributor tests (from other model configurations) as needed
- if tests fail, contacting the appropriate Team Lead to have the issue addressed
- serving as gatekeeper for the trunk/master branch, in consultation with the Lead Coordinator

Team Leads are responsible for coordination of the work required of each Task Team and for cross-team coordination. They are appointed by the Executive Oversight Board and report to the Lead Coordinator. They ensure timely completion of each Team's work and serve as primary points of contact for communications with the Lead Coordinator and other Team Leads. A committee

consisting of the Team Leads, Lead Coordinator, Co-lead Coordinator and Software Engineer acts as a science oversight/advisory board and also functions as a change control board for the code.

Coordination roles within the Consortium include

- Having the authority to make decisions isolated to their teams
- Assigning Team personnel, coordinating with other Team Leads as needed (e.g. when they do not control funding for the task)
- Coordinating with each other to prevent conflicts
- Setting up limited-term working groups and appointing their leaders to further explore particular strategies and/or technical choices, as needed

Planning roles:

- Ensuring that the most useful developments in the research and operational communities are integrated into the software base
- Coordinating developments planned by scientists outside the Consortium with the Consortium [frequency] Plan
- Reviewing trends and emerging science and technologies to identify and prioritize opportunities for advancing the capabilities of the model for use by the research and operational communities

Task Team members are responsible for the planning and implementation of tasks needed to maintain and distribute the model software and documentation for Consortium Members and the community in a manner that promotes the ease of future model development. Team members may be from multiple institutions and are appointed by and report to the Team Leads. They make suggestions for best practices and work with developers to carry them out. Team members' work includes

- Writing scripts for testing and analysis tools
- Incorporating new developments (scientific or technical)
- Performing basic software testing and oversight per best practices
- Providing assistance to new users
- Providing practical support for user meetings

Task Team members are not responsible for support of scientific research and development, parameter tuning or emergency responses following operational failures.

Current Task Teams are

- CICEdyn
- Icepack
- Infrastructure
- Repository

- Testing and Analysis
- Community Support

Workshop participants provided nominations for Executive Oversight Board representatives: Dave Bader (ACME), Bill Large (NCAR), Ruth Preller (NRL), Hendrik Tolman (NWS), [TBD] (UKMO), Wieslaw Maslowski (NPS), [TBD] (ECCC).

Volunteers for Team Leads: Lead Coordinator (Hunke), Icepack ([DOE TBD], Winton), CICEdyn (Lemieux - Dynamics, Roberts - Transport), Analysis & Testing (Allard), Repository (Software Engineer, Turner [committee lead]), Community Support (Bailey, Blockley). Tony Craig was suggested for Software Engineer.

Consortium Membership

It is necessary to distinguish between sea ice model developers and Consortium Members in order to define and limit decision-making authority and responsibilities. Care should be taken to emphasize that the purpose of the Consortium is to enhance sea ice model development through community collaboration with few time-consuming demands on code users.

An explicit minimum FTE level was not defined for determining Consortium Membership, initially. Such a limit could prevent institutions supported by soft money from participating, since explicit mention of required FTE time in proposals might be problematic for some reviewers. Also, the FTE level needed to complete Consortium work likely will change over time, once the organization and processes are set up and running. In the future, it is possible that serious community code contributors could become Consortium Members.

For the next two years, Consortium Members will make the following in-kind contributions (FTE fractions, * indicates they are looking for people to hire or otherwise fill the position):

- DOE 0.5 *
- NCAR 1.0 *
- Environment Canada 0.25
- GFDL 0.25
- NOAA (in addition to GFDL) 0.5 *
- NRL-Stennis 0.5 *
- NPS 0.25
- UK Met Office 0.5 plus possibly an additional 0.5 from University of Reading

If a Consortium Member's representative for a critical role must change, then the Consortium Member should give more than two weeks notice and recommend (via their EOB member) another person for the vacancy. If a Consortium Member can no longer participate at all, the Lead Coordinator shall convene the Team Leads committee to discuss and decide how to cover the gaps, in consultation with the

EOB. If a Member leaves the Consortium, its code cannot be removed from the Consortium repository.

Policies and guidance

Policy and guidance needs were articulated during the workshop, to be fully defined and agreed upon afterward. Exploration of the associated issues and further discussions will be led by the Lead Coordinator. Topics include

- Intellectual property protection for code sharing and publications
- What other formal agreements are necessary among Consortium and community members?
- Process for institutional acceptance of new code contributions (see testing)
- Process and timeline for nominations/appointments of leadership and team members
- Succession/backup plan for leadership roles

Required reporting

Workshop participants recommend that required documentation and reporting of Consortium activities be limited to once a year. The Lead Coordinator will compile an update for the Executive Oversight Board with input from the Team Leads, otherwise soliciting verbal updates from Team Leads in the meantime.

Settling disputes

If a dispute cannot be settled at the Task Team level, then the committee of Team Leads will handle discussion of the issue, with input from Lead Coordinator. The Lead Coordinator has the authority to make binding decisions, but may take particularly difficult issues to the EOB for its advice and assistance.

Code contribution responsibilities

When contributing new code, both non-Consortium community members and Consortium Members must run and pass the test suite themselves. New code must be accompanied by associated documentation describing the change and its effect, i.e. both testing and analysis results.

Planning

Tasks for the next year are associated with setting up the Consortium. For future planning, Team Leads and the Lead and Co-Lead coordinators will track community members' interests in order to prepare for and coordinate those contributions to the code. The planning horizon will need to accommodate the typical three-year proposal cycle for new research, with the goal of rapidly transferring new capabilities into operational use. Consortium planning exercises should also incorporate operational plans and timelines to the extent possible. Workshop participants recommended against imposing fixed timelines with hard deadlines.

Code Management

Constraints

The Consortium needs to concentrate on the sea ice model code itself (initially CICE), including the physics modules, infrastructure, testing and analysis software, and related capabilities such as coupling interfaces and data assimilation code. The Consortium's work does not include tuning and validation of the model in other model settings such as coupled climate and operational configurations, because of the sea ice model's high sensitivity to atmospheric and oceanic forcing.

The Consortium will provide a basic, common test suite and analysis package for evaluating changes to the model. Test cases will verify that a change does not break the code, while analysis tools quantify simulation improvement or degradation in the context of the stand-alone configuration. Regression tests should be completely automated and easy to execute. Analysis tools exist at individual modeling centers and could be put in a library available through the Consortium for everyone to use. Similarly, publications and other validation information from the modeling centers should be put in the repository for ease of sharing.

Some parts of the current code are unique to individual Consortium Members, such as coupled model interfaces (drivers, coupling fields), and analysis packages. The interfaces can be included in the repository to simplify the incorporation of necessary changes originating in other parts of the CICE code. Task Teams are responsible for making changes in all interfaces (including Icepack) related to their model changes, with oversight from the Software Engineer. Each Consortium Member is responsible for testing its own interfaces, packages, and configurations. Software packages that are very Member-specific may not be maintained by Consortium, e.g. if they are useful only for a single Member.

Software guidelines will be written, providing further constraints on code development, maintenance and management. Software contributors will be responsible for running the standard test suite and analysis package as the first step of the code acceptance process. To the extent possible, their contributions should abide by the principle of "do no harm."

Process

Figure 2 displays a conceptual framework for code management and review, including pathways for the development process, which was presented for consideration by workshop participants. It captures the idea of increasing code review and testing requirements as the code advances through stages of wider use to its public release, and it assumes a hierarchy of access permissions for different versions of the code. Depending on the outcome of post-workshop explorations of institutional requirements for intellectual property protection and source code

access, the repository might function a little differently than is illustrated in this framework.

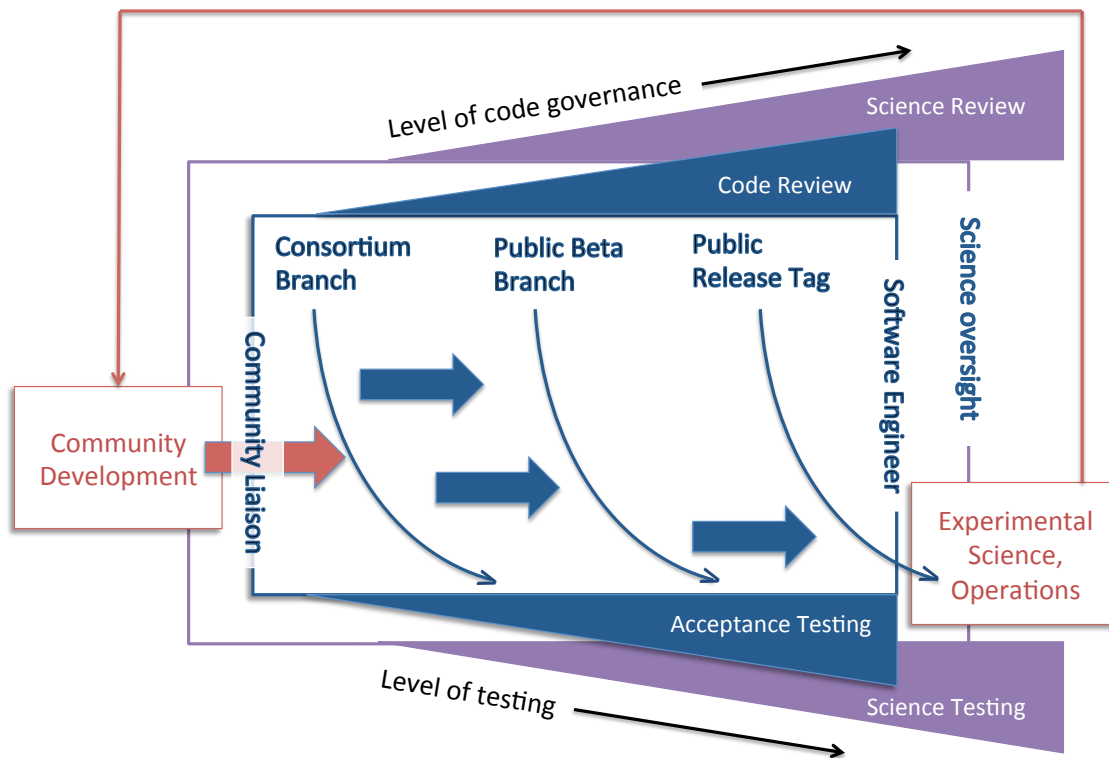


Figure 2. Code Management Design. The Consortium is represented in blue, with community involvement in red. Purple represents community testing and use, in which Consortium Members participate through their own research or operational programs, with the committee of Team Leads providing oversight and coordination of community contributions for the Consortium.

The Software Engineer's and Community Liaison's roles encircle the Consortium, representing broad oversight and the initial interface for community involvement, respectively. Here, R&D contributions from outside the Consortium membership ranks would begin with a publicly released version of the code, although Consortium Members and alpha or beta testers with access to more up-to-date versions of the code would be encouraged to use those instead. At each stage, new code contributions must pass certain tests and reviews.

The repository can be considered as a set of branches with varying access permission levels, which can be merged or forked as needed. There will be a primary branch for Consortium Members' and alpha testers' use, called the Consortium master branch, which is in practice a development branch for working toward a public software release. As confidence in the code's reliability increases through testing (see below), that code may be made available as a public beta

branch with an “at your own risk” caveat, for finding and fixing bugs prior to a tagged public release.

While the Team Leads review the code throughout the process, the Software Engineer, Lead Coordinator and Co-lead Coordinator have a gatekeeping responsibility for the master branches. If the Consortium uses Git for the repository, private development branches can be created on local computing platforms by anyone with access to the master branches (not shown).

As active Consortium Member representatives, the Team Leads are responsible for their institution’s code acceptance testing and scientific experiments. The committee of Team Leads also forms a science oversight committee responsible for identifying new scientific and technical capabilities for future incorporation in the code, a role that extends beyond the Consortium itself into the community. Because scientific research and development is not part of the Consortium’s mission, it will not direct science decisions made by the community and its funding sources. However, as a hub for integration of new sea ice model developments, the Consortium can assist in coordination of these developments among community members, e.g. by recommending collaborative partnerships to reduce duplication of effort. Team Leads will then know what new major contributions to expect and be better able to prioritize and coordinate tasks within the Consortium. This symbiotic relationship between the community and the Consortium has been the essential ingredient contributing to CICE’s past success.

The public release and the master branches for the Consortium and public beta version will be supported by the Consortium. Older versions of the code will not be supported, other than bug fixes in the current public release.

Outdated or unused portions of the software likely will need to be pruned from the code base in the future. All Consortium Members will need to provide input for pruning decisions, with the final decision made by the committee of Team Leads.

Code releases

Workshop participants recommend that the software be released on a regular schedule (twice a year) to a beta-tester group, with a public release every two years. If there are insufficient code updates to warrant a new beta release, that cycle can be skipped, but the most valuable approach is to encourage developers and beta testers to update to the master branches every few months, in order to find bugs more quickly. Versions used by operational centers need to be publicly available.

Confidence Testing

It is impossible to centralize all scientific and code acceptance testing for the model, because its configuration and performance is highly dependent on the coupled configurations in which it is used. A set of basic, mandatory test cases will be

designed and released for general use by both Consortium Members and the broader community.

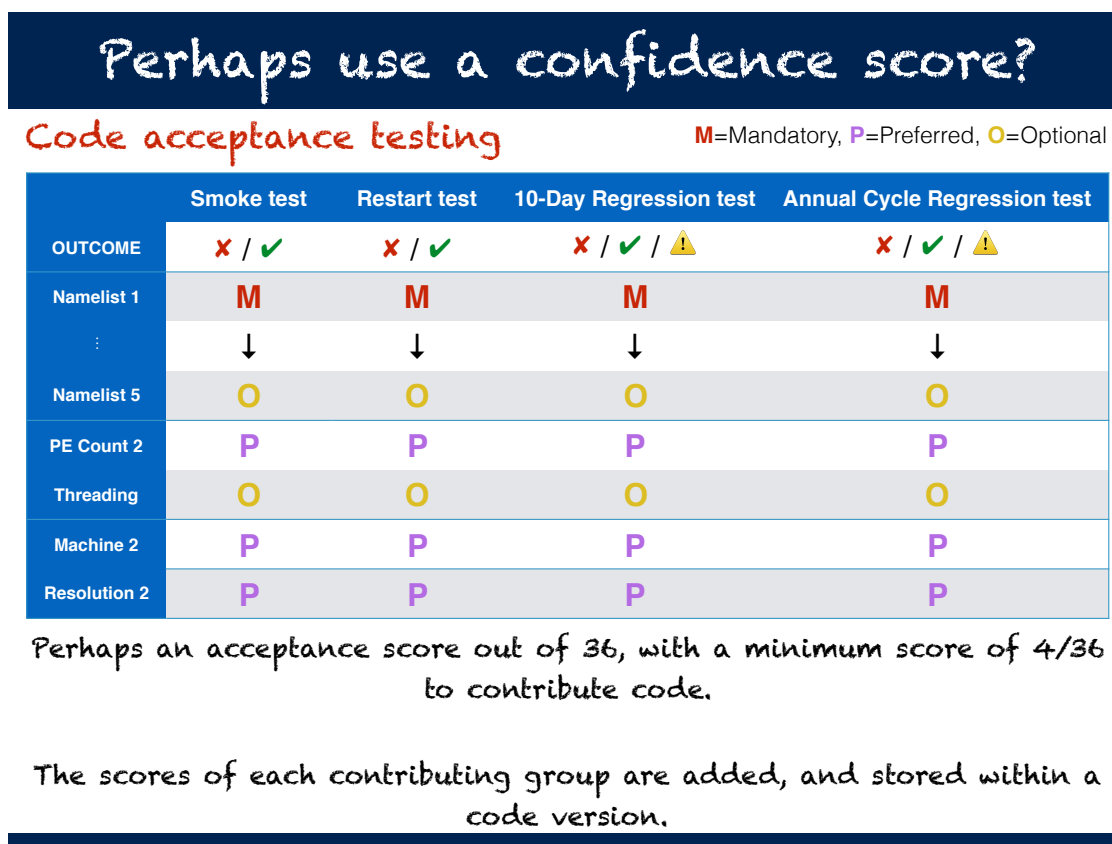


Figure 3. Confidence testing involves a standard set of basic tests (e.g. restart, regression) performed for a set of different sea ice model configurations (e.g. namelist options, parallelization options, and different resolutions). Each Consortium Member and beta tester has the opportunity to run any or all of these tests, contributing the resulting pass/fail results to an overall confidence score. In this example, a minimum score of 4 (passing the mandatory tests) would be required for the code to be contributed to the Consortium.

In addition, a more complete testing mechanism will be designed that includes progressively more comprehensive tests (Figure 3). As more Consortium Members and beta testers run these tests, pass/fail results can be compiled and stored as part of the model version in the repository, resulting in a confidence score for that version. A minimal confidence score based on the mandatory test cases is required for code to be contributed to the Consortium. As it grows, the confidence score reflects further testing in different coupled model settings by various Consortium Members and beta testers as well as positive results from a variety of tests. When the confidence score reaches a pre-defined threshold, the code is deemed acceptable for further scientific acceptance testing and use.

The Consortium cannot enforce usage of particular release or beta versions, particularly for operational centers whose timelines for code updates involve extensive acceptance testing of their own. Continuously and flexibly building confidence levels in code versions has the potential to minimize the amount of time a code version spends in the beta-testing stage, by allowing Consortium Members unable to immediately test the code to abstain from doing so.

Scientific acceptance testing will need to include stand-alone, ice-ocean, regional coupled, and global coupled configurations as well as alternative options used by some centers such as Data Assimilation. The Consortium itself can only support the stand-alone configuration, therefore Consortium Members must test the code in their own configurations. The CESM and NEMO coupled modeling efforts already have standard configurations that are scientifically validated, against which sea ice model changes can be tested at those modeling centers.

The principle of “Do no harm”

Many new code contributions can be made modular and isolated via namelist or preprocessing flags. When such code is added to the code base, the simulation results must be bit-for-bit identical to results prior to the addition when the new code is turned off. If the change can not be turned off and changes the results, then the reason for the change must be pinpointed and documented in the repository logs for that version (e.g. “the order of operations changed at line N”).

Next Steps

Consortium organization

1. Document workshop outcomes (Lead Coordinator)
Create a basic web page for the workshop on the current CICE trac site, pending an improved website design (currently expected to be part of the repository). Put workshop agenda, presentation slides on the website.

Complete this report with input from workshop participants and post it on the website.
2. Invite Executive Oversight Board members to join the effort (Lead Coordinator)

Nominations: Dave Bader (ACME), Bill Large (NCAR), Ruth Preller (NRL), Hendrik Tolman (NWS), [TBD] (UKMO), Wieslaw Maslowski (NPS), [TBD] (ECCC).¹

3. Document governance/process protocols (Lead Coordinator and Team Leads)

Explore solutions for the associated issues and present a strawman for further discussion among Team Leads. Topics include

- Intellectual property protection for code sharing and publications, including code redistribution
- What other formal agreements are necessary among Consortium and community members?
- Process for institutional acceptance of new code contributions (see testing)
- Process and timeline for nominations/appointments of leadership and team members
- Succession/backup plan for leadership roles

4. Get formal agreements in place, if needed (Lead Coordinator, EOB, Team Leads)

Draw up agreements with software developers or users, specifying expectations and requirements for its use prior to public release.

Specify policy for alpha and beta testers, addressing co-authorship for publications, use (or not) of contributed intellectual-property code, and a process for notifying users of sensitive code and primary contact information.

5. Consortium Members hires (Team Leads)

Identifying and hiring a software engineer is a top priority, to enable his/her input during the initial setup of Consortium processes and repository. (NWS 0.5 FTE)

NRL, NCAR and LANL also plan new hires or other personnel actions to fulfill their roles.

Code

1. Decide on and set up repository (Repository Committee, SE, Team Leads)

¹ Jean-Francois Lemieux agreed to serve on the EOB during the workshop, but we later decided it would be better if he were not both a Team Lead and an EOB member.

A repository committee was formed during the workshop whose members are currently:

Adrian Turner (LANL, chair), Dave Bailey (NCAR/CESM), Andrew Roberts (NPS), Hendrik Tolman (NOAA/NWS), Ligia Bernardet (NOAA/DTC), Ed Blockley (UKMO), Alan Wallcraft (NRL), Mike Winton (NOAA/GFDL), Tony Craig (contractor).

Their task is to investigate potential options and evaluate those options in terms of the Consortium's requirements, many of which were articulated during the workshop. The committee will report back to the Team Leads with responses for the following items and other points raised during their investigation:

- a. identify what is essential to get the repository structure to work for the Consortium
 - b. leverage what centers are already doing
 - c. define what we mean by open source, private branches
 - d. describe automated testing tools
 - e. describe bug tracking software
 - f. describe how the "seamless interface" works
 - g. can anyone who can get to your code use it?
 - h. if CICE is open source, must every code that links with it also be open source?
 - i. propose a repository structure
 - j. determine options for a web presence
 - k. administration: cost? can different institutions have 'root' access?
2. Determine the starting code version (Lead Coordinator, Team Leads)

This issue is complicated by the fact that various groups are currently using different CICE release or Icepack versions. Discussion will need to take into consideration their timelines for upgrading to the latest version, or willingness to deal with two versions simultaneously (their current one and the latest one). The Consortium will officially launch with a public release of the CICE code with Icepack, but the scope of the release needs to be determined (test suite, analysis tools, etc). LANL will need to coordinate with the ACME Council for permission to extend the current CICE collaboration agreement to new collaborators and for an open-source repository.
3. Set up regression testing (Testing and Analysis Team with input from all teams)

4. Define software best practices (SE, Team Leads)
5. Describe workflow examples for changes, new code contributions (Lead Coordinator, Team Leads)
6. Compile work plan/prioritization/task assignments (Lead Coordinator, Team Leads)

Each team will have a number of tasks required to get the code management structure set up and the code itself ready for release. For instance, the Testing and Analysis Team needs to create a standard test suite and begin putting together a metrics/validation package. The other teams will need to help develop tests pertinent for their parts of the code. For example, the Testing and Analysis and Icepack teams can develop a driver with specific tests for Icepack. Centers that already have a test suite and validation package will provide substantial help for this process.

A backlog of contributed code tasks will need to be addressed, some of which can be used as test cases for developing the code management process.

The Infrastructure Team will need to initiate planning for integration of data assimilation, C grid, and other new capabilities desired by the Consortium.

Initial Timeline

By early December:

- draft workshop report circulated, near final
- Next meeting via GoToMeeting teleconferencing

by end of 2016 calendar year:

- identify, hire software engineer
- form Executive Oversight Board based on nominations

by the end of January (3 months)

- have initial planning document put together from Team Leads, including repo committee, governance/process/policy issues
- begin interviews for new hires

Appendix A: Acronyms

CPO – Climate Program Office
DoD – U.S. Department of Defense
DOE – U.S. Department of Energy
ECCC – Environment and Climate Change Canada
EMC – Environmental Modeling Center
ESRL – Earth Systems Research Laboratory
GFDL – Geophysical Fluid Dynamics Laboratory
GLERL – Great Lakes Environmental Research Laboratory
LANL – Los Alamos National Laboratory
NGGPS – Next Generation Global Prediction System
NOAA – National Oceanic and Atmospheric Administration
NPS – Naval Postgraduate School
NRL-Stennis – Naval Research Laboratory, Stennis Space Center
NWS – National Weather Service
OAR – Ocean & Atmospheric Research
OWAQ – Office of Weather & Air Quality
STI – Science Technology Integration
UK – United Kingdom

Appendix B: Participant List

DOE	Dorothy Koch*	Dorothy.Koch@science.doe.gov
DOE/LANL	Elizabeth Hunke	eclare@lanl.gov
DOE/LANL	Nicole Jeffery*	njeffery@lanl.gov
DOE/LANL	Bill Lipscomb	lipscomb@lanl.gov
DOE/LANL	Adrian Turner	akt@lanl.gov
DOE/LANL	Wilbert Weijer	wilbert@lanl.gov
DoD/NPS	Andrew Roberts	afrobert@nps.edu
DoD/NRL-Stennis	Rick Allard	richard.allard@nrlssc.navy.mil
DoD/NRL-Stennis	Pam Posey	pamela.posey@nrlssc.navy.mil
DoD/NRL-Stennis	Ruth Preller	ruth.preller@nrlssc.navy.mil
DoD/NRL-Stennis	Alan Wallcraft	Alan.Wallcraft@nrlssc.navy.mil
NOAA/NWS/EMC	Robert Grumbine	Robert.Grumbine@noaa.gov
NOAA/NWS/STI	Hendrik Tolman	Hendrik.Tolman@noaa.gov
NOAA/NWS/STI	Karen Keith	Karen.Keith@noaa.gov
NOAA/OAR/CPO	Jin Huang	Jin.Huang@noaa.gov
NOAA/OAR/ESRL-NGGPS	Ligia Bernardet	Ligia.Bernardet@noaa.gov
NOAA/OAR/ESRL	Amy Solomon	Amy.Solomon@noaa.gov
NOAA/OAR/GFDL	Mike Winton	Michael.Winton@noaa.gov
NOAA/OAR/GLERL	Jia Wang	jia.wang@noaa.gov
NOAA/OAR/OWAQ	Jessie Carman	jessie.carman@noaa.gov
NSF	Bill Wiseman	wwiseman@nsf.gov
NSF/NCAR	Dave Bailey	dbailey@ucar.edu
NSF/NCAR	Marika Holland	mholland@ucar.edu
NSF/NCAR/NGGPS	Louisa Nance	nance@ucar.edu
NSF/NCAR/NGGPS	Don Stark	stark@ucar.edu
ECCC	Jean-Francois Lemieux	Jean-Francois.Lemieux@canada.ca
UK Met Office	Ed Blockley	ed.blockley@metoffice.gov.uk

* by phone

Appendix C: Governance/Process FAQ

What authority do the Leads, including the Lead Coordinator have?

Team leads are responsible for making sure their team's work is carried out. Their agency/institution, by appointing them through the Executive Oversight Board, gives them the authority to assign people and resources to work on it. When other institutions are responsible for some of their team's work, Team Leads may need to apply pressure via the Lead Coordinator and the Executive Oversight Board, if they are unsuccessful themselves.

How will decisions get made within and across the Consortium?

The Team Leads make decisions that affect only their teams. Cross-team decisions are discussed among the Team Leads, with final decision authority resting with the Lead Coordinator. The Executive Oversight Board may decide certain significant questions (see the description of its roles).

How will the Consortium prioritize developments?

This will be the Lead Coordinator's decision, in consultation with the Team Leads, as part of the planning process.

If changes happen in one Team that would affect work in another, will there be cross-group discussion? Will there be periodic meetings of the Team Leads to discuss this sort of thing?

Yes.

What if a development does not perform well in a particular model?

Most new developments can be made optional in the code. For more invasive changes, the Executive Oversight Board may need to decide what is done, if the Lead Coordinator and Team Leads cannot reach consensus.

What happens if a team or team-member is not delivering?

Team Leads and the Lead Coordinator work with them to understand the problem and find solutions. If this is unsuccessful, the Lead Coordinator may take the problem to the Executive Oversight Board, which can apply additional pressure within agencies/institutions. If this is unsuccessful, then the Executive Oversight Board may take the issue to the Sponsor level.

Will new software or model developments be made available to the general modeling community immediately?

We currently expect that most developments can be made available on a fully open branch of the repository, without delay. Consortium Members may have dual roles, as Consortium Members maintaining the code and as members of the user community contributing new model developments; rules for code release could hinge on this distinction. For example, as users, institutions wishing to be more cautious in releasing their new developments to the public may hold them back temporarily, while still making them available to other Consortium members.